MADYMO

Coupling Manual

Release 7.2

January 2010

© Copyright 2010 by TASS
All rights reserved.

MADYMO® has been developed at TASS BV.

This document contains proprietary and confidential information of TASS. The contents of this document may not be disclosed to third parties, copied or duplicated in any form, in whole or in part, without prior written permission of TASS.

The terms and conditions governing the license of MADYMO® software consist solely of those set forth in the written contracts between TASS or TASS authorised third parties and its customers. The software may only be used or copied in accordance with the terms of these contracts.

MADYMO Manuals

An overview of the MADYMO solver related manuals is given below. From Acrobat Reader, these manuals can be accessed directly by clicking the manual in the table below. Manuals marked with a star (*) are also provided in hard-copy (major releases only).

Theory Manual	The theoretical concepts of the MADYMO solver.
Reference Manual*	Detailed information on how to use the MADYMO solver and how to specify the input.
Model Manual*	Dummy, Dummy Subsystem and Barrier Models with simple examples.
Applications Manual	Example applications using Dummy, Dummy Subsystem and Barrier Models.
Human Model Manual	Human Models and applications that make use of Human Models.
Tyre Model Manual	Documentation about Tyre Models.
Transition Manual	Describes the translation from MADYMO 5.4 format (DATA) to MADYMO 6.0 format (XML), including a description on how to handle Dummy Models in the conversion. A full conversion table is available.
Utilities Manual	User's guide for MADYMO / Optimiser, MADYMO / Scaler, MADYMO / Dummy Generator, MADYMO / Tank Test Analysis, MADYMO / XML Translator, MADYMO / XML Reformatter and MADYMO / XML Expander.
Folder Manual	Describes the use of MADYMO / Folder.
Programmer's Manual	Information about user-defined routines.
Release Notes	Describes the new features, modifications and bug fixes with respect to the previous release.
Installation Instructions	Description for the system administrator to install MADYMO.
Coupling Manual	Description of the direct coupling with ABAQUS, LS-DYNA, PAM CRASH / SAFE and Radioss and the TCP / IP coupling with MATLAB / Simulink.

TASS Offices

For questions and support on using MADYMO, please contact our office or local representative in your region or consult our website at www.tass-safe.com.

HEAD OFFICE

TASS HQ
Schoemakerstraat 97
2628 VK Delft
The Netherlands
Telephone: +31 888 277 000
Fax: +31 888 277 003
Email (information): info.nl@tass-safe.com

North America

TASS Americas
38701 Seven Mile Road, Suite 260
Livonia, MI 48152
USA
Telephone: +1 734 779 4850
Fax: +1 734 779 4858
Email (support): support.na@tass-safe.com
Email (information): info.na@tass-safe.com

South America

m.v^2 engenharia
R. Pedroso Alvarenga, 1046
Cj.:48-04531-012
Itaim Bibi
São Paulo SP
Brazil
Telephone: +55 11 3892 8680 / 5774
Fax: +55 11 3892 8683
Email (support): support.br@tass-safe.com
Email (information): info.br@tass-safe.com

Germany, Austria, Switzerland, Eastern Europe

TASS Germany
Paulinenstraße 1
65189 Wiesbaden
Germany
Telephone: +49 711 21 843 108
Fax: +49 711 89 244 540
Email (support): support.de@tass-safe.com
Email (information): info.de@tass-safe.com

France, Spain, Italy, Portugal

TASS France
Immeuble Le Fujy
1 Rue du Centre
93160 Noisy Le Grand
France
Telephone: +33 1 7061 8335
Fax: +33 1 7997 4891
Email (support): support.fr@tass-safe.com
Email (information): info.fr@tass-safe.com

UK, Ireland, Benelux, Scandinavia

Advanced Simulation Technologies Ltd
The Loft
Kings Lane
Stratford Upon Avon
CV37 0RD
United Kingdom
Telephone: +44 17 8926 4782
Fax: +44 17 8929 7765
Email (support): support.uk@tass-safe.com
Email (information): info.uk@tass-safe.com

India

Cax Software Private Ltd
G-1, M S Crystal
12, Malleshpalya Main Road
Bangalore 560075
India
Telephone: +91 80 411 51 512
Fax: +91 80 411 51 511
Email (support): support.in@tass-safe.com
Email (information): info.in@tass-safe.com

China

TASS China
2301 Sino Life Tower
No.707 Zhangyang Road, Pudong
200120 Shanghai
China
Telephone: +86 21 5835 8036 * 800
Fax: +86 21 5836 0838
Email (support): support.cn@tass-safe.com
Email (information): info.cn@tass-safe.com

Korea

TASS Korea
Works C502, Garden 5,
289, Munjeong-dong, Songpa-gu
Seoul, 138-200
Korea
Telephone: +82 70 8650 5220
Fax: +82 50 5115 5220
Email (support): support.kr@tass-safe.com
Email (information): info.kr@tass-safe.com

Japan

TASS Japan
Nisso No. 15. Building 3F
2-17-19 Shin-Yokohama Kohoku-ku
Yokohama 222-0033
Japan
Telephone: +81 45 473 7955
Fax: +81 45 473 7959
Email (support): support.jp@tass-safe.com
Email (information): info.jp@tass-safe.com

Australia, New Zealand

Advea Engineering Pty Ltd
7 Calderwood Ave
Wheelers Hill
Vic. 3150
Australia
Telephone: +61 3 9558 6666
Fax: +61 3 9558 6677
Email (support): support.au@tass-safe.com
Email (information): info.au@tass-safe.com

Contents

MADYMO Manuals iii

TASS Offices iv

1 General Information 1

1.1 Introduction . . . 1

1.1.1 Why Coupling? . . . 1

1.1.2 Coupling Type . . . 1

1.1.3 Simulation Control . . . 3

1.2 Platforms and versions . . . 3

1.3 Program Installation . . . 4

1.4 Performance Guidelines . . . 4

1.5 Limitations and Known Issues . . . 5

2 New Features 7

3 Interface Version 10

3.1 Introduction . . . 10

3.2 Features of the 6.3.1 interface and onwards . . . 10

3.2.1 Identifing the Interface . . . 10

3.3 Interface History . . . 11

4 Madymo Input Deck Setup 12

4.1 MADYMO input deck setup for Basic Coupling . . . 12

4.2 MADYMO input deck setup for Extended Coupling . . . 13

4.3 Troubleshooting . . . 17

5 Coupling with ABAQUS 18

6 Coupling with LS-DYNA **19**

6.1 Introduction . . . 19

6.2 Simulation Control . . . 19

6.3 LS-DYNA Input Deck Setup . . . 19

6.3.1 Basic Coupling . . . 20

6.3.2 Extended Coupling . . . 22

6.4 Limitations and Known Issues . . . 23

7 Coupling with PAM CRASH/SAFE **25**

7.1 Introduction . . . 25

7.2 Simulation Control . . . 25

7.3 PAM CRASH/SAFE Input Deck Setup . . . 25

7.3.1 Basic Coupling . . . 25

7.3.2 Extended Coupling . . . 28

7.4 Limitations and Known Issues . . . 30

8 Coupling with Radioss **31**

8.1 Introduction . . . 31

8.2 Simulation Control . . . 31

8.3 Radioss Input Deck Setup . . . 31

8.3.1 Basic Coupling . . . 32

8.3.2 Extended Coupling . . . 33

8.4 Restart . . . 35

8.4.1 Restart Control . . . 35

8.5 Limitations and Known Issues . . . 36

9 Example Startup Scripts **37**

9.1 PAM CRASH/SAFE . . . 37

9.2 LS-DYNA . . . 37

9.3 MPP-DYNA 38
9.3.1 MPP-DYNA with lammpi 38
9.3.2 MPP-DYNA with mpich/native mpi 38
9.3.3 MPP-DYNA with mpi that doens't propagate env. vars 38
9.3.4 MPP-DYNA with LSF on IBM 39
9.4 Radioss 39
9.4.1 Radioss smp 39
9.4.2 Radioss mpp 40
9.4.3 Radioss mpp and restart 40

10 MATLAB/Simulink coupling **41**
10.1 Compatible MATLAB version 41
10.2 Interfacing MADYMO with MATLAB/ Simulink 42
10.3 Using MATLAB/ Simulink in combination with MADYMO 43
10.4 Conditions 45
10.5 Miscellaneous 46

1 General Information

1.1 Introduction

This manual gives an overview of the requirements to perform a simulation with MADYMO coupled to another program. Additionally, it explains why MADYMO facilitates a coupling with other software, how the coupling works and what types of coupling are supported. Hereafter, the coupling to another program is generically termed PARTNER, unless otherwise specifically indicated. All information in this section applies to all partners unless otherwise specified.

1.1.1 Why Coupling?

A coupling to another software product provides the opportunity for the user to combine, in a single simulation, the features from both products. The user may need to combine the two products because some features may be available in one product and not in the other or the user may simply rely on one software product for one type of calculation and on the other product for other types of calculations.

1.1.2 Coupling Type

The coupling between two software products is done through a coupling interface. The information to be send from one software product to the other is first gathered internally. This information is then send to the PARTNER through the coupling interface routines. The PARTNER distributes the data, performs calculations and subsequently gathers data and sends is back through the interface. This process is illustrated in Fig. 1.1.

Depending on the type of data that is communicated, various coupling types can be identified:

- MB object data is communicated; at each time integration point, displacements are sent from MADYMO to the PARTNER code, which in turn sends forces acting upon these objects back to MADYMO. This is called *Basic Coupling*.

  In basic coupling, the PARTNER program builds a copy of the MB object natively in MADYMO. This copy allows the PARTNER program to put forces acting on the MB object, and to display it in its kinematic output file.

- FE data is communicated; at each time integration point, nodal displacements

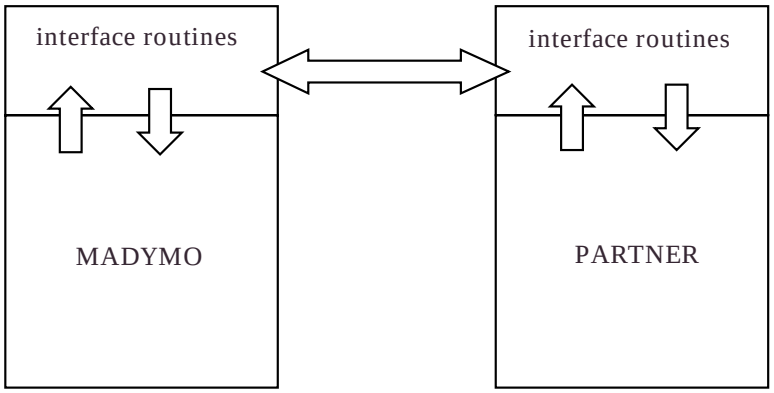


Figure 1.1: Illustration of the communication through the coupling interface.

are sent from the PARTNER program to MADYMO, which in turn sends forces acting on these nodes back to the PARTNER program. This is called the *Extended Coupling*.

In extended coupling, MADYMO builds a copy of (part of) the FE model it receives from the PARTNER program. Now MADYMO can put forces on the nodes, and display the FE model in the kinematic output file (KIN3 file).

- It is also possible to mix these two couplings. This is called *Combined Coupling*.

The reprint file reflects the coupling type under the — COUPLING — section.

All coupling types synchronize their time step, so the minimum of the PARTNER and MADYMO time step is taken.

When MADYMO dummy models are used in coupling it is recommended to use the extended coupling. This is because the compliance of the dummy contact surfaces is not taken into account in the basic coupling. However, the basic coupling generally runs more efficiently, because less data needs to be communicated between MADYMO and the PARTNER.

Please be aware that this MADYMO release supports both basic and extended coupling, but some PARTNER codes do not. For more information, see their respective sections.

1.1.3 Simulation Control

- The simulation end time has to be defined both in PARTNER and MADYMO. When defined differently in both, the simulation will end when the lowest end time is reached.
- The kinematic output time step in the MADYMO input file and the parameters of the option in the PARTNER input file must be consistent. The combined output will be out of phase, otherwise.
- Define the number of CPU's in the MADYMO input deck. If you plan to run an SMP job, the number of CPU's used by MADYMO and PARTNER must be equal. MADYMO defines the number of CPUs, say #n, by NR_PROC in the XML input file with

```
<CONTROL_ALLOCATION
    NR_PROC  = "#n"
/>
```

- Generally, command line options cannot be used in coupling simulations. For example, -nrproc (to set the number of CPU's), -isize, -rsize, -csize (to set memory allocation) and others cannot be used. Please use the input decks to specify the required information.[1]

1.2 Platforms and versions

In order to run a coupling simulation, one needs both the special coupling executable as well as the MADYMO shared library. Starting from MADYMO 6.3, the latter is distributed by default with each MADYMO release / patch on all supported Unix and Linux platforms (please refer to the Installation Instructions). The special executable from the PARTNER software may be obtained from the PARTNER offices.

With the introduction of the MADYMO shared library, the user can choose which MADYMO version he wants to use with a certain PARTNER version, and vice versa. To do so, he simply uses a newer version of MADYMO. If set up correctly, the PARTNER executable will pick up the new shared library.

For a list of validated MADYMO–PARTNER combinations, please refer to the MADYMO Release Notes.

[1] There is one exception: specifying the number of CPU's when coupling with PAM CRASH / SAFE, see Sec. 9.1

1.3 Program Installation

This section describes the steps needed for installation in order to perform a coupling simulation.

1. Install MADYMO according to the MADYMO Installation Instructions. If MADYMO is already installed, go to step 3.
2. Install the MADYMO license file *madymo.lic* for MADYMO.
3. Verify that the MADYMO license for coupling to the PARTNER is present in the MADYMO license file.
4. Install PARTNER software according to their installation instructions, including, if needed, licenses required by PARTNER for using the coupling with MADYMO.
5. Install the special coupling executable which is to be obtained from the PARTNER.
6. Add the path to the **madymo72** run script to the PATH environment.

1.4 Performance Guidelines

Starting from MADYMO 6.3, all MADYMO shared libraries are SMP. This implies that all coupling simulations are SMP enabled. For SMP-SMP calculations, in general more CPU's will result in a faster runtime (up to 8 CPU's). The performance of an SMP-MPP calculation depends both on the machine(s) on which the job is run and on the input deck. For the SMP-MPP coupling it is strongly recommended to run MADYMO on 1 CPU.

Note that coupling simulations in which MADYMO runs in MPP mode are not yet supported.

- The amount of data that is to be communicated, invariably has an impact on the performance. Besides being communicated twice, the data is also gathered and distributed twice every time step. To increase performance, please have a close look at the data that is communicated an remove any unneeded or unused data.
- The user is advised to check the MADYMO Timing Information in the log and / or reprint file. It will give an indication of the CPU ratio that each party takes in

the coupling calculation. MADYMO typically takes 5 - 15% of the CPU time. Increasing MADYMO performance will not give good over-all speed improvements. For example, when MADYMO uses 10% of the CPU time, even halving the MADYMO CPU time will only yield an overall speed increase of 5%. Significant speed improvements are generally to be found in the party which uses most of the CPU time.

- Especially for SMP-MPP calculations, have a look at the CPU / Wall Clock ratio. In general, this should be higher than 90%. Low ratios can be caused by (among others)
  - Allocating more nodes than available CPU's.
  - Bad network connections
  - Wrong MPI version (especially for Linux)
  - Sending large amounts of data over the coupling interface

  If in doubt, please contact your local support office. Please note that MADYMO does not use any MPI calls, so any problems caused by it can best be handled by the PARTNER support office.
- When the user is in doubt if the coupling is having a bad impact on the performance, please run the coupled executable as if it was a stand-alone executable. All coupled executables provided by PARTNER, can be run in stand-alone mode. By disabling coupling (both in keywords in the input deck and command line options), all influences of the coupling interface are removed and the most objective comparison between coupling and standalone can be made. Please keep in mind that for best comparison, all MADYMO data must be replaced by the PARTNER counterparts.

1.5 Limitations and Known Issues

- Restarting a coupled simulation is only supported with Radioss.
- Sometimes the PARTNER program writes "NORMAL TERMINATION", even when MADYMO terminates abnormally.

  Therefore the user is advised to check the following messages after running a coupled simulation:
  - In the screen output[2], PARTNER writes "NORMAL TERMINATION" (or a similar message) if no error appears on the PARTNER side.

[2] Not all partners write there output such that it ends up in the log file

  - The message: "MADYMO TERMINATED NORMALLY" must appear in the log file and reprint file.

- For extended coupling, it is required to send at least one element from PARTNER to MADYMO. It is not allowed to send only nodes.
- Planes attached to a body are incorrect positioned if the center of gravity is nonzero.
- When performing an SMP-MPP job, the output to standard out (typically the screen), and therefore the MADYMO log file, can be out of phase. This is normal behavior. On some systems, this behavior can be changed, see the MPI documentation for more information.
- On AMD Opteron systems (linux24-x86_64 and linux26-x86_64), MADYMO 6.3.1 and higher are incompatible with PARTNER executables that run with MADYMO 6.3. The job stops with the following error:

  ```
  <cplexe>: error while loading shared libraries: <cplexe>: undefined symbol:
  pghpf_sect1
  ```

  where <cplexe> is the PARTNER coupling executable.

  The solution is to update the PARTNER executable to a version build with PGI compiler version 6.1.

2 New Features

MADYMO 7.2 has new features related to coupling.

- R7.1:
  - Constraint of a jet to external FE nodes (see Sec. 4.2)
- R7.0:
  - For shell elements, the combination of more than one element type (i.e. TRIAD3 and QUAD4 elements) in one PART of the external FE partner has been made possible for the coupling analyses. This can be seen in the following example, in which PART 1 of the external FE partner consists of two TRIAD3 elements (i.e. element 1 and 2) and three QUAD4 elements (i.e. element 3, 4 and 5):

```
*ELEMENT_SHELL
          1          1          1          4          2
          2          1          2          4          5
          3          1          2          5          6          3
          4          1          4          7          8          5
          5          1          5          8          9          6
```

  - Constraint of MADYMO rigid body and external FE node(s) (see Sec. 4.2)
  - Constraint of MB belt to external FE node (see Sec. 4.2)
  - Constraint of MADYMO nodes and external FE element

    If the slave (MADYMO) node of a TIED_SURFACE.* lies in the gap of the master (PARTNER) element, it is put on the master element. This can be undesired. An update has been performed in the TIED_SURFACE models so the offset of the slave node with respect to the master element can be maintained if desired.

    Here follows an example. The driver airbag (MADYMO) is "supported" to the (PARTNER) steering wheel by means of TIED_SURFACE.BREAK_-FORCE.

```
<TIED_SURFACE.BREAK_FORCE
   ID="1"
   NAME="tied_surface1"
   MASTER_GROUP_LIST="/External_SW_gfe"
   SLAVE_GROUP_LIST="/DAB_sup_gfe"
   SLAVE_OFFSET="ON"
   GAP_TYPE="VALUE"
   GAP_VALUE="4.0E-1"
   MAXFN="2E6"
   MAXFS="2.8E6"
   GMODE1="2E1"
   GMODE2="2E1"
   WINDOW="1.0"
/>
```

– Global positioning

  INITIAL.REF_SPACE facilitates positioning of the reference space with respect to the global space. In this way the entire MADYMO model can be repositioned with respect to an external FE model in a coupled simulation.

  An example in which the origin of the reference space is translated to (1.0 1.0 0.0) in the global space.

```
<INITIAL.REF_SPACE
   POS = "1.0 1.0 0.0"
/>
```

– Solver output scaling

  Specifying a time scale factor (TIME_SCALE_FACTOR_ANI) in the MADYMO input file, will enable the animation output in a different time unit (e.g. seconds) than the current one (e.g. milliseconds). In output format MAD the time is by default in ms (TIME_SCALE_FACTOR_ANI="1"), in the other formats (HDF5, D3PLOT and H3D) in s. TIME_SCALE_FACTOR_ANI="FACTOR" means that the time values in the animation output files are multiplied by the factor FACTOR compared to the default. The time scale factor enables the time in the animation output in different units (e.g. seconds), which facilitates overlay of animation files for coupling in post-processors. For example using TIME_SCALE_FACTOR_ANI="1000" when writing out animation output in D3PLOT format, results in the animation output time being multiplied by a factor 1000, thus effectively the time is written out in ms. The time scale factor affects the animation, contour and Gasflow output in the following formats: MAD, HDF5, D3PLOT and H3D.

```
<CONTROL_OUTPUT
   FILTER_IGNORE = "OFF"
   EXTENDED_SAMPLING_IGNORE = "OFF"
   SCALE_FACTOR_ANI = "1.0"
   TIME_SCALE_FACTOR_ANI = "1000"
   TIME_STEP = "1.0E-04"
   TIME_STEP_ANI = "2.0E-03"
   TIME_STEP_RESTART = "1.0E-02"
   WRITE_DEBUG = "NONE"
   WRITE_FEMESH = "OFF"
   >
   <ANIMATION
    WRITE_FORMAT = "D3PLOT"
   />
   <TIME_HISTORY_MB
      BODY_OUTPUT_LIST = "/Dummy_sys/Pelvis_acc"
   />
   <TIME_HISTORY_ENERGY
      ENERGY_OUTPUT_LIST = "AirbagEnergy_out"
   />
</CONTROL_OUTPUT>
```

– Smaller step than LIMIT_STEP allowed in coupling

A smaller FE time step than LIMIT_STEP is allowed only for an external (COUPLING) FE_MODEL. Due to this a coupling run will continue if the external program require (temporary) a smaller time step. E.g. 1.0E-07 in the example below.

```
<CONTROL_FE_TIME_STEP
   MIN_STEP = "1.0E-06"
   MAX_STEP = "1.0E-05"
   LIMIT_STEP = "1.0E-07"
   NR_OF_CYCLES = "1"
/>
```

- R6.3.2:
  - The MADYMO command line interface can now generate an environment file (`.madymo_env`) with the command line option `-show`. This should be done in the same directory as where MADYMO will be run. The MADYMO executable will read the environment settings it needs from this file when it cannot find the settings in the environment variables. This makes it easier to run MADYMO without the command line interface, or when the MPI version used for the PARTNER executable has problems in communicating environment variables.

    Please note that the value of `LD_LIBRARY_PATH` (or `SHLIB_PATH` on `ibmrs51` platforms) cannot be communicated through this file. A bad `LD_LIBRARY_PATH` setting prevents MADYMO from starting and hence prevents reading this environment file.

    The following environment variables are stored in the `.madymo_env` file:

    ```
    MADETCPATH
    MADINCPATH
    MADBINPATH
    ```

  - Restarting is supported with Radioss 51G (see Sec. 8.4).

3 Interface Version

3.1 Introduction

With the introduction of MADYMO 6.3.1, a coupling interface is used. This interface has new features, yet provides full backwards compatibility to enable seamless integration with older PARTNER executables that do not have this interface build-in.

3.2 Features of the 6.3.1 interface and onwards

The interface provides two features:

Automatic Kinematic Output Scaling MADYMO can scale its kinematic output to the length units of the PARTNER. To enable this, set the AUTO_SCALE_ANI keyword under COUPLING to "ON". The default behaviour is "OFF", to enable backwards compatibility.

Communication of failure exit to PARTNER codes MADYMO sends an error termination flag to the PARTNER code, such that it can write an abnormal termination message to the screen. Also, the PARTNER code can send such a flag to MADYMO, causing MADYMO to write an abnormal termination message to the log and reprint file.

For both features applies that the PARTNER executable must support this, meaning that they must also use the interface.

3.2.1 Identifing the Interface

When MADYMO notices that the PARTNER executable is using an older interface, MADYMO will not use these features and fall back on a version that the PARTNER software does support. In this way, the largest common version of the interface is used. This version is the *effective interface version.* The reprint file reflects this version under the — COUPLING INTERFACE — section.

When this *effective interface version* is not reflected in the reprint file, an older version of MADYMO (version 6.3 or earlier) is used. These versions are always build with interface version 0, and therefore do not have the abovementioned improvements.

3.3 Interface History

This section reflects what MADYMO releases are build with which interface versions, newest first. For a detailed list of supported and tested platforms, consult the MADYMO public download area at www.tass-safe.com.

Version	Release	Description
1	6.3.1 and onwards	Automatic Kinematic Output Scaling Error Termination Flag
0	6.3	Interface as used with MADYMO version 6.3

4 Madymo Input Deck Setup

In order to enable coupling to an external executable, the MADYMO input deck must contain the <COUPLING/> element. Note that one must always select INT_MTH = "EULER" under CONTROL_ANALYSIS.TIME for a coupling simulation.

4.1 MADYMO input deck setup for Basic Coupling

This section describes the elements that must be added to the MADYMO input deck for a Basic Coupling simulation.

Note that basic coupling is not supported in the coupling with ABAQUS.

- Rigid bodies which are to be coupled to PARTNER are referred to under the COUPLING element as shown in the following example:

```
<COUPLING>
  <COUPLING_BODY
    EXTERNAL_REF ="1"
    BODY ="/1/1"
    EXTERNAL_DATA ="163"
  />
</COUPLING>
```

  - The values in EXTERNAL_REF and EXTERNAL_DATA are send to PARTNER. For more details, please refer to Sec. 7.3.1 for basic coupling to PAM CRASH / SAFE or Sec. 8.3.1 for basic coupling to Radioss.
  - Bodies cannot be coupled to LS-DYNA.

- Ellipsoids and planes which are to be coupled to PARTNER are defined under the COUPLING element as shown in the following example:

```
<COUPLING>
  <COUPLING_SURFACE
    EXTERNAL_REF="3"
    SURFACE="/2/1"
    EXTERNAL_DATA="9"
  />
</COUPLING>
```

  - The values in EXTERNAL_REF and EXTERNAL_DATA are send to PARTNER. For more details, please refer to Sec. 6.3.1 for basic coupling to LS-DYNA, Sec. 7.3.1 for basic coupling to PAM CRASH / SAFE or Sec. 8.3.1 for basic coupling to Radioss.

– Cylinders cannot be coupled to LS-DYNA, PAM CRASH/SAFE and Radioss.

The elements <COUPLING_BODY/> and <COUPLING_SURFACE/> may be combined within the same <COUPLING/> element.

4.2 MADYMO input deck setup for Extended Coupling

This section describes the XML elements that should be added to the MADYMO input deck for an Extended Coupling simulation.

When performing an extended coupling simulation, the amount of data transferred between PARTNER and MADYMO can make it necessary to increase the double precision and integer space in MADYMO, depending on the model sizes. Therefore, the user may have to increase the double precision and integer space of MADYMO using the I_SIZE and R_SIZE settings in the MADYMO XML input deck under CONTROL_ALLOCATION.

In the unlikely event that MADYMO complains about the character space allocated, also increase the C_SIZE setting.

It is recommended to perform a so called 'zero-run' first (with the end time identical to the start time). Inspect the MADYMO reprint file (between the lines COUPLING_INTERFACE and END COUPLING_INTERFACE) to check that MADYMO reads the data correctly from PARTNER. Also check any errors and warnings reported by either MADYMO or PARTNER.

Topology of (degenerated) elements can be transferred from PARTNER to MADYMO. Elements of type TRIAD3, QUAD 4 and HEXA8 are allowed to be sent over.

1. Add an empty FE_MODEL to the MADYMO input file (under SYSTEM.*) to represent the PARTNER model that will be coupled to MADYMO. Within MADYMO, all data received from the PARTNER is assembled in a single FE model. For example:

```
<SYSTEM.MODEL
   ID="2"
   NAME="Ext_System"
   >
   <FE_MODEL
      ID="1"
      NAME="Ext_FE_Model"
      >
      <CONTROL_FE_MODEL
```

```
      />
      <CONTROL_FE_TIME_STEP
      />
   </FE_MODEL>
</SYSTEM.MODEL>
```

2. Add a COUPLING element with a reference to this FE model:

```
<COUPLING
   FE_MODEL="/Ext_System/Ext_FE_Model"
/>
```

3. With the previous steps, the MADYMO side has been prepared to receive data from the external FE-model. The coupled FE model can be used similar to a native MADYMO FE model, with the restriction that no kinematic constraints can be used. Specifically, kinematic contact, spot welds, prescribed motion and rigid elements cannot be defined in the MADYMO input deck for the external FE model. Next, the user must define what to do with this data. Four examples follow.

   (a) Contact Interaction

   If an external group is to be used in a contact calculation, add the following:

   i. Define a specific GROUP_FE elements for the PARTNER elements involved in the contact interaction. Groups of elements, nodes and / or parts can be used.

```
<GROUP_FE
   ID="2"
   NAME="Ext_Group_gfe"
   FE_MODEL  ="/Ext_System/Ext_FE_Model"
   PART_LIST ="ALL"
/>
```

   The attribute PART_LIST may also be combined or replaced by other lists, for example a NODE_LIST and / or ELEMENT_LIST (see the MADYMO Reference Manual for details). The value ALL may also be replaced by more specific part, node or element numbers provided that the PARTNER input deck specifies that these values are communicated to MADYMO. Please note that numbering of nodes and elements of the PARTNER FE coupled parts in MADYMO may not be the same as in the PARTNER FE input deck.

   ii. Define a contact between this group and the MADYMO model groups using CONTACT.FE_FE or CONTACT.MB_FE. Here follow two examples:

```
<CONTACT.FE_FE
      ID="21"
      MASTER_SURFACE="/ES2_lhs/Pelvis_gfe"
      SLAVE_SURFACE="RADIOSS_door_gfe"
```

```
      >
      <CONTACT_METHOD.NODE_TO_SURFACE_CHAR>
         <CONTACT_FORCE.CHAR
            CONTACT_TYPE="MASTER"
            FRIC_FUNC="Friction03_fun"
         />
      </CONTACT_METHOD.NODE_TO_SURFACE_CHAR>
   </CONTACT.FE_FE>
```

and

```
   <CONTACT.FE_FE
      ID="1"
      MASTER_SURFACE="/Ext_Coupling_gfe"
      SLAVE_SURFACE="/DAB_gfe"
      >
      <CONTACT_METHOD.SURFACE_TO_SURFACE>
         <CONTACT_FORCE.ADAPTIVE
            DAMP_COEF="0.05"
            TIME_STEP="2.05E-6"
         />
         <GAP_TYPE.FUNC
            GAP_FUNC="/Gap_fun"
         />
         <CONTACT_EDGE/>
         <INITIAL_TYPE.CHECK/>
      </CONTACT_METHOD.SURFACE_TO_SURFACE>
   </CONTACT.FE_FE>
```

Please note the following:

- All contact types can be used (elastic and penalty based) except kinematic contact.
- Damping in the contact(s) between MADYMO and PARTNER can be used.
- For contact between a PARTNER FE model and MADYMO MB models use CONTACT.MB_FE.

(b) Connections Defining Tied Surfaces

If (part of) a MADYMO FE model is to be connected to (part of) an external FE model, add a tied surface. The constraints between the MADYMO surface and the external surface can be defined using TIED_SURFACE.SPOTWELD, TIED_SURFACE.BREAK_FORCE / STRESS. Groups in the coupled FE model should be used for the MASTER_GROUP_LIST. For the SLAVE_GROUP_LIST, groups in other FE_MODEL elements in the MADYMO input file should be used. The nodes of the slave group will be tied to the PARTNER master surface.

```
<TIED_SURFACE.SPOTWELD
   ID="96"
   NAME="Spotweld_Cpl"
   SLAVE_GROUP_LIST="spotweld_gfe"
   MASTER_GROUP_LIST="Ext_Group_gfe"
   GAP_TYPE="VALUE"
   GAP_VALUE="0.1"
/>
```

(c) Constraint of MADYMO rigid body and external FE node(s)

SUPPORT_RESTRAINT has been added, based on a point-restraint model. In this way external FE nodes can be constrained to MADYMO rigid bodies.

In the following example the nodes 101:103 of the external FE model "Coupling/Vehicle_part" are supported on the MADYMO rigid body "/SYS1/Cube_bod". The stiffness and damping characteristics are computed based on the time-step and the damping characteristic according to the critical damping because no user CHARACTERISTIC.LOAD is specified.

```
<SUPPORT_RESTRAINT
   BODY = "/SYS1/Cube_bod"
   FE_MODEL = "/Coupling/Vehicle_part"
   NODE_LIST = "101:103"
/>
```

(d) Constraint of MB belt to external FE node

POINT_OBJECT.MB being referred from a BELT_SEGMENT now facilitates connecting a belt segment to an external coupled finite element model. Here follows an example which illustrates the usage.

```
<BELT_SEGMENT
   ID="4"
   NAME="belt4_Buc_Part"
   POINT_REF_1="belt2_Buc"
   POINT_REF_2="belt4_Buc"
   CHAR="char_132"
   INITIAL_STRAIN="5.000000E-05"
   ADD_LENGTH="0.0"
/>
<POINT_OBJECT.MB
   ID="3"
   NAME="belt2_Buc"
   FE_MODEL="/Coupling_Sys/Ext_Fem"
   NODE="4"
   />
<POINT_OBJECT.MB
   ID="8"
   NAME="belt4_Buc"
   FE_MODEL="/Seat_Sys/Shoulder_Belt_Fem"
   NODE="77"
/>
```

(e) Constraint of a jet to external FE nodes

FE_CRDSYS under JET.* (except JET.GAS_FLOW_ELEMENT) now facilitates connecting a jet to an external coupled finite element model. An example which illustrates the usage:

```
<SYSTEM.MODEL
  ID="10"
  NAME="coupling_system.model"
>
  <FE_MODEL
    ID="1"
```

```
      NAME="coupling_fe_model"
      >
      <CONTROL_FE_MODEL/>
      <CONTROL_FE_TIME_STEP/>
      <FE_CRDSYS.NODE
        ID="1000"
         NAME="fe_jet_sup_external"
         NODE1="8"
         NODE2="9"
         NODE3="5"
       />
.....................
.....................
  </FE_MODEL>
</SYSTEM.MODEL>
.....................
.....................
    <JET.IDELCHIK
        ID="1"
        FE_CRDSYS="/10/1/1000"
        CENTRE="0.19574338 0.19574338 0.19574338"
        OUTFLOW_DIR="1.0 0.0 0.0"
        >
        <JET_SHAPE.RECTANGULAR
          LONG_LENGTH="0.01"
          SHORT_LENGTH="0.01"
          LONG_DIR="0.0 1.0 0.0"
        />
     </JET.IDELCHIK>
```

4.3 Troubleshooting

The user is advised to check the following topics when having trouble in getting the simulation to run properly:

- Look at all warnings and errors, both from MADYMO as well as the PARTNER program.
- Make sure that the data send from the PARTNER program is in SI units. Some partners allow non-SI units to be used. Use the appropriate conversion factors to force sending all data in SI units.
- When running SMP-MPP jobs: is the MPI environment installed correctly and working OK?
- View the kinematic output from both MADYMO and the PARTNER on top of each other. Have a close look at the data that is send to the other program: is it moving simultaneously (and as expected) in both output files? If not, you probably put some kinematic constraints on the data that is communicated. Identify these constraints and remove them.

5 Coupling with ABAQUS

For further information, please contact your local support agency. Coupling with Abaqus is in use by dedicated customers.

6 Coupling with LS-DYNA

6.1 Introduction

This section gives specific information related to the coupling with LS-DYNA. All information is applicable to both the SMP as well as the MPP version of LS-DYNA (MPP-DYNA), unless specifically indicated otherwise.

Please note that the details of the input deck for LS-DYNA have been based on LS-DYNA 970 and may be subject to modifications. Always refer to the latest LS-DYNA documentation for the correct syntax.

6.2 Simulation Control

- For general simulation control, please refer to Sec. 1.1.3.
- Defining the number of CPUs / Nodes to run on:
  - For an SMP job, LS-DYNA uses the number of CPUs defined under the *CONTROL_PARALLEL keyword in the input file.
  - For an MPP job, MPP-DYNA, the number of nodes is defined through the MPI command line option (see Sec. 9.3).

6.3 LS-DYNA Input Deck Setup

This section describes the required modifications to the LS-DYNA input deck to perform Basic Coupling or Extended Coupling simulations. See the LS-DYNA User Manual for details on the cards not mentioned here specifically. Modifications to the MADYMO input deck are described in Sec. 4.

For all couplings, add the *CONTROL_COUPLING card to set the units relationship between MADYMO and LS-DYNA:

```
*CONTROL_COUPLING
$#  UNLENG    UNTIME    UNFORC    TIMIDL     FLIPX     FLIPY     FLIPZ    SUBCYL
     1000.        1.        1.        0.         0         0         0         0
```

Here the 1000 specifies that LS-DYNA should divide their length values by 1000 before sending them to MADYMO. Length values received from MADYMO will be multiplied by 1000. This control coupling card example shows that the LS-DYNA length

units are: mm. The time and force numbers are unchanged with respect to MADYMO (so second's and Newton's).

6.3.1 Basic Coupling

- Define parts using *PART for representing the MADYMO ellipsoids and planes. This has to be repeated for each ellipsoid and plane in MADYMO.
- Define shells for parts representing the MADYMO ellipsoids and planes using *SECTION_SHELL.
- Define rigid body material behavior for parts representing MADYMO ellipsoids and planes using *MAT_RIGID:

```
*MAT_RIGID
$#     MID        RO         E        PR         N    COUPLE         M  RE/ALIAS
```

  - RE / ALIAS must match the EXTERNAL_REF number of the MADYMO ellipsoid or plane in the COUPLING data block (see Sec. 4.1). RE / ALIAS must be unique for all coupling entities.
  - EXTERNAL_DATA is not used by LS-DYNA.
  - COUPLE: coupling flag. Set to 2 to generate a mesh in LS-DYNA for MADYMO ellipsoids and planes and write these in the d3plot output file.
  - M and N are ignored. These parameters are used for compatibility with the MADYMO 5.4.1 coupling.
  - Beware of correct unit conversion scales of R_0 and E values when defining MAT_RIGID.

- Define contacts for the parts representing MADYMO ellipsoids or planes using *CONTACT_ENTITY. Some coupling related cards are:

  - GEOTYP (card1):
    6: Plane
    7: Ellipsoid
  - SO (card2): Contact stiffness type:
    0: Rigid MADYMO ellipsoid or plane
    1: Contact stiffness to be taken from E value in *MAT_RIGID
    2: User-defined force-penetration curve
  - G1 (card5): ID for MADYMO ellipsoid or plane

- Optional: LS-DYNA nodes can be supported on a MADYMO rigid body using *CONSTRAINED_EXTRA_NODES_NODE. This option can be used to connect an airbag model or belt anchorage points in LS-DYNA to a steering wheel model in MADYMO, for example.

Input Example Basic Coupling

The following is part of a MADYMO input deck for Basic Coupling:

```
<COUPLING>
   <COUPLING_SURFACE
      EXTERNAL_REF="1"
      SURFACE="/2/1"
      EXTERNAL_DATA="0.0"
   />
</COUPLING>
```

This will send data from surface /2/1 to LS-DYNA. In LS-DYNA one may refer to this data using reference 1. Note the arbitrary value 0.0 for EXTERNAL_DATA, which is not used in LS-DYNA.

The following is part of a LS-DYNA input deck for Basic Coupling:

```
*MAT_RIGID
         2 7.82000E+09 2.06800E+05 0.3000000         1         2         1         1
         0
*SECTION_SHELL
         1           0         0.0       0.0       0.0       0.0
       1.0         1.0         1.0       1.0       0.0
*PART
$  Part-ID  Section-ID Material-ID
      9001           1           2
```

Verification

The user is advised to check the log file and the reprint file to verify if all entities are communicated correctly. The LS-DYNA log file shows the LS-DYNA output of all entities received from MADYMO, the reprint file shows the entities MADYMO sends to LS-DYNA.

The LS-DYNA log file:

```
    MADYMO3D external number           = 1.00000E+00
    MADYMO3D coupling flag............ = 2.00000E+00
```

More information below these lines is printed in the LS-DYNA log file concerning the

FE mesh builds on top of the ellipsoid. All of this information should be checked.

The accompanying MADYMO reprint file:

```
 COUPLING
    FE_MODEL ....................................:
      SURFACE ...................................: /2/1
    EXTERNAL REFERENCE ........................:         1
    EXTERNAL DATA .............................:
    NUMBER ..:          1  DATA ..:      0.0000E+00
```

6.3.2 Extended Coupling

- Add the LS-DYNA *SET (*SET_PART, *SET_SHELL, *SET_SOLID) cards to define the LS-DYNA element sets to be used in the coupling contact interaction.
- Add the *CONTACT_COUPLING card to define the LS-DYNA sets (just defined) to be used for the contact in MADYMO. The format of *CONTACT_COUPLING is:

  ```
  *CONTACT_COUPLING
     ID
     SID
     STYPE
  ```

  where ID is the contact identifier; SID is the identifier of the SET to be used for coupling and STYPE is the SET type, which can have the following values:

  0 : part set (default)
  1 : shell element set
  2 : solid element set
  3 : thick shell set

Input Example Extended Coupling

The following is part of a MADYMO input deck.

```
   <COUPLING
      FE_MODEL="/10/3"
   />
```

The following is part of a LS-DYNA input deck.

```
*CONTACT_COUPLING
      98
      63
*SET_PART
      63
      92 93
```

This forces LS-DYNA to send all elements belonging to part 92 and 93 to MADYMO.

Verification

The user is advised to check the log file and the reprint file to verify if all entities are communicated correctly. The log file shows the LS-DYNA output of all FE data send to MADYMO, the reprint file shows the FE data that MADYMO received from LS-DYNA.

Example LS-DYNA log file:

```
c o n t a c t   c o u p l i n g

contact coupling interface ID     =      98
number of part IDs                =       1
number of shell element IDs       =       0
number of solid element IDs       =       0
number of solid-shell element IDs =       0

shell element ID list for coupling:
        1         2         3         4

input summary including part IDs:
number of part IDs                =       9
number of unique nodal points     =       1
number of shell elements          =       4
number of solid elements          =       0
number of solid-shell elements    =       0
```

Accompanying reprint file:

```
COUPLING
   FE_MODEL ..............................: /10/3 ( /coupling_model/fe_model )
NUMBER OF NODES                          :          9
NUMBER OF PARTS                          :          1
  PART ID (FROM COUPLED PROGRAM)         :         92
    THICKNESS                            :      1.0000E-01
    BULK MODULUS                         :      2.5000E+07
NUMBER OF ELEMENTS                       :          4
  NUMBER OF TRIADS                       :          0
  NUMBER OF QUADS                        :          4
  NUMBER OF SOLIDS                       :          0
```

6.4 Limitations and Known Issues

- When the MADYMO end time is higher than the LS-DYNA end time, MADYMO can write the terminate line twice to the log file, with the first line:

  MADYMO TERMINATED NORMALLY.

  and the second line:

  MADYMO TERMINATED ABNORMALLY, because of ERRORS

The second line should not be written to the screen and log file, and is misleading.

In the reprint file, there is only one termination line and it should read:

MADYMO TERMINATED NORMALLY.

Workaround: set the two end times identical to each other.

- When the initial time step of LS-DYNA is larger than that of MADYMO, the simulation can get out of sync.

  Workaround: specify a smaller initial timestep than the MADYMO timestep, by changing the value of DTINT under *CONTROL_TIMESTEP in the LS-DYNA input file.

7 Coupling with PAM CRASH/SAFE

7.1 Introduction

This section gives specific information related to coupling with PAM CRASH / SAFE. All current implementations of coupling with PAM CRASH / SAFE are SMP only.

Please note that the details of the input deck for PAM CRASH / SAFE have been based on PAM CRASH / SAFE 2005 and may be subject to modifications. Always refer to the latest PAM CRASH / SAFE documentation for the correct syntax.

7.2 Simulation Control

- For general simulation control, please refer to Sec. 1.1.3.
- PAM CRASH / SAFE reads the number of CPU's it uses from the environment variable OMP_NUM_THREADS. This number must match the number of CPU's used by MADYMO, otherwise the simulation will terminate.

7.3 PAM CRASH/SAFE Input Deck Setup

This section describes the required modifications to the PAM CRASH / SAFE input deck to perform Basic Coupling or Extended Coupling simulations. See the PAM CRASH / SAFE User Manual for details on the cards not mentioned here specifically. Modifications to the MADYMO input deck are described in Sec. 4.

7.3.1 Basic Coupling

For Coupling either a coupling body or a coupling surface can be defined.

Body Coupling

- The value defined in EXTERNAL_DATA refers to the node that will be tied with the body center of gravity. In PAM CRASH / SAFE, this node can be defined as free or connectivity node. Post-processing of this node can be done in PAM CRASH / SAFE.

Surface Coupling

- EXTERNAL_DATA contains 2 values, i.e. EXTERNAL_DATA="5 3". The first value defined in EXTERNAL_DATA is used in PAM CRASH / SAFE to refer to part id's. Only in the ellipsoid case, the second value corresponds to the discretization density.

Define a PART and MATERIAL (type 100) section for all surfaces to be received from MADYMO. The PART identifiers should match to the first value of the accompanying EXTERNAL_DATA value in the MADYMO input deck. Note that PART and MATERIAL cards can be re-used. For further details, please refer to the PART SECTION and MATERIALS SECTION in the PAM CRASH / SAFE manual.

Input Example Basic Coupling

The following two examples are parts of a MADYMO input deck for Basic Coupling.

Body Coupling Example

```
<COUPLING>
   <COUPLING_BODY
      BODY = "/Vehicle/HUB"
      EXTERNAL_REF = "10"
      EXTERNAL_DATA = "30729"/>
</COUPLING>
```

The following is part of a PAM CRASH / SAFE input deck for Basic Coupling:

```
NODE  /    30729       2.7479956       0.3514169      0.82089777       0
SHELL /    30023     143   30720   30721   30729   30728       3 0.0
SHELL /    30024     143   30721   30722   30730   30729       3 0.0
SHELL /    30031     143   30728   30729   30737   30736       3 0.0
SHELL /    30032     143   30729   30730   30738   30737       3 0.0


$
```

Surface Coupling Example

```
<COUPLING>
   <COUPLING_SURFACE
      EXTERNAL_REF = "10"
      SURFACE = "/1/1"
      EXTERNAL_DATA = "1.0 3.0"/>
</COUPLING>
```

EXTERNAL_REF is ignored, although it is mandatory.

This will send data from surface /1/1 to PAM CRASH/SAFE. In PAM CRASH/SAFE one may refer to this data using reference 1. The density of the mesh created for this surface is 3.

The following is part of a PAM CRASH/SAFE input deck for Basic Coupling:

```
$
PART  /        1SHELL          1
NAME ELASTIC MEMBRANE
       0.0

 3.5000e-4

END_PART
MATER /        1     100           1000.       0       0       1       0
               0       0       0       0       0       0      1.       0
NAME ELASTIC MEMBRANE
  1000000.               0.35




$
```

Verification

The user is advised to check the log file and the reprint file to verify if all entities are communicated correctly. The log file shows the PAM CRASH/SAFE output of all entities received from MADYMO, the reprint file shows the entities MADYMO sends to PAM CRASH/SAFE.

The log file:

```
       *** SETUP MODE SUCCESSFULLY READ FROM MADYMO **


      NUMBER OF COUPLED ENTITIES .........         1
      NUMBER OF COUPLED ELLIPSOIDS .......         1
      NUMBER OF TIED NODES  ..............         0
      NUMBER OF PLANES ...................         0

      COUPLING ENTITY NO .................         1 (ELLIPSOID)
      SYSTEM NUMBER ......................         1
      ELLIPSOID NUMBER ...................         1
      ASSIGNED MATERIAL NUMBER (TYPE 100).         1
      AUTOMATIC MESH GENERATION DENSITY ..         3
      NUMBER OF GENERATED NODES ..........        92
      NUMBER OF GENERATED NULL SHELLS ....        90

      TOTAL NUMBER OF GENERATED NODES.....        92
      TOTAL NUMBER OF GENERATED SHELLS....        90
```

The accompanying MADYMO reprint file:

```
---------------------------------- COUPLING ----------------------------------

 COUPLING
    FE_MODEL ....................................:
      SURFACE ...................................: /2/1
    EXTERNAL REFERENCE ..........................:         10
    EXTERNAL DATA ...............................:
    NUMBER ..:          1  DATA ..:      1.0000E+00
    NUMBER ..:          2  DATA ..:      3.0000E+00

------------------------------- END COUPLING ---------------------------------
```

7.3.2 Extended Coupling

- Define the MDBODY cards; please refer to the PAM CRASH / SAFE Solver Reference Manual for detailed information. Briefly, the MDBODY card contains the following entities

  ```
  $---5---10----5---20----5---30----5---40----5---50----5---60----5---70----5---80
  $       IDMDBO  CTYPE   BULK    THICK   IDCVS--|
  MDBODY/
  $   TITLE
  NAME
  ```

  where the relevant control parameters for the coupling with MADYMO are

  - IDMDBO, i.e. the FE deformable body identification number
  - CTYPE, i.e. the FE mesh type (SHELL, SOLID or NODE)
  - BULK, i.e. the average bulk modulus to be used by MADYMO.
  - THICK, i.e. the average thickness to be used by MADYMO.
  - IDCVS, i.e. external identifier to be referenced in MADYMO.
  - Every MDBODY card is followed by cards to define element or node selection by keywords (See General Entity Selection sub-section in the CONTROL SECTION of the PAM CRASH / SAFE manual)

- The "COUPLING MADYMO" keyword is mandatory in the control section.
- This card should be repeated for each FE Deformable Body to be sent to MADYMO.
- If the average bulk modulus and / or the average thickness are not defined, average values will be computed by PAM CRASH / SAFE.
- The identifiers IDCVS, set in the MDBODY cards, can be referred to in the MADYMO input deck as part id's.

- If MDBODY type equals NODE, the IDCVS label can not be used in MADYMO. Instead, one must refer to the individual node numbers. Note that element and part information related to the communicated nodes is not send to MADYMO.

Input Example Extended Coupling

The following is part of a MADYMO input deck.

```
<COUPLING
   FE_MODEL="/Coupling_System/External_Boule"
/>
```

The following is part of a PAM CRASH / SAFE input deck.

```
$---5---10----5---20----5---30----5---40----5---50----5---60----5---70----5---80
$       IDMDBO  CTYPE   BULK    THICK   IDCVS--|
MDBODY/        2SHELL                          3
NAME boule
        PART 1
        END
```

This forces PAM CRASH / SAFE to send all shell elements belonging to part 1 to MADYMO. Average bulk modulus and thickness are calculated by PAM CRASH / SAFE. In MADYMO the elements can be referred to either by element number or by the reference number 3(!).

Verification

The user is advised to check the log file and the reprint file to verify if all entities are communicated correctly. The log file shows the PAM CRASH / SAFE output of all FE data send to MADYMO, the reprint file shows the FE data that MADYMO received from PAM CRASH / SAFE.

Example log file:

```
 F E   B O D Y   F O R   M A D Y M O


     FE DEFORMABLE BODY ID ............ =           2
     FE DEFORMABLE BODY NAME .......... = boule
     BODY MESH TYPE ................... = SHELL
     REFERENCE ID IN MADYMO CONTACT ... =           3
     AVERAGE BULK MODULUS ............. =  0.1111E+10
     AVERAGE THICKNESS ................ =  0.3500E-03


     NUMBER OF ASSOCIATED SHELLS ...... =          90
```

```
              LIST OF SHELLS TRANSFERED TO MADYMO:

 ====  DATA SUPPRESSED BY NOPRINT OPTION  ====
```

Accompanying reprint file:

```
----------------------------COUPLING INTERFACE------------------------------
 NUMBER OF NODES                       :         92
 NUMBER OF PARTS                       :          1
   PART ID (FROM COUPLED PROGRAM)      :          3
     THICKNESS                         :      3.5000E-04
     BULK MODULUS                      :      1.1111E+09
 NUMBER OF ELEMENTS                    :         90
   NUMBER OF TRIADS                    :          0
   NUMBER OF QUADS                     :         90
   NUMBER OF SOLIDS                    :          0
```

7.4 Limitations and Known Issues

- The last DSY file might not be written if it occurs at the MADYMO simulation end time point (you can slightly increase the MADYMO simulation end time to avoid this problem).

8 Coupling with Radioss

8.1 Introduction

This section gives specific information related to the coupling with Radioss.

Please note that the details of the input deck for Radioss have been based on Radioss 4.7.4 and may be subject to modifications. Always refer to the latest Radioss documentation for the correct syntax.

8.2 Simulation Control

- For general simulation control, please refer to Sec. 1.1.3.
- Only the simulation end time point defined in the MADYMO input file is considered. The end time point defined in the Radioss file is ignored.
- The /MODIF Radioss's option is not suppported while using coupling.
- To enable coupling, the option /MADYMO/ON in the Radioss ENGINE input file is necessary.

```
/MADYMO/ON
Lunit Tunit Munit
```

Lunit	Length unit conversion factor, must be 1 meter in Radioss model units
Tunit	Time unit conversion factor, must be 1 s in Radioss model units
Munit	Mass unit conversion factor, must be 1 Kg in Radioss model units

```
/MADYMO/ON
1   1000  1
```

Here the 1000 specifies that Radioss should divide their time values with 1000 before sending them to MADYMO (and the inverse for all time numbers received from MADYMO), effectively stating that the Radioss input file is in ms. The lenght and force numbers are unchanged.

8.3 Radioss Input Deck Setup

This section describes the required modifications to the Radioss input deck to perform Extended Coupling or Basic Coupling simulations. See the Radioss User Manual for

details on the cards not mentioned here specifically. Modifications to the MADYMO input deck are described in Sec. 4.

8.3.1 Basic Coupling

- Note that Radioss only uses the value of EXTERNAL_REF. However, the attribute EXTERNAL_DATA is mandatory in MADYMO and must therefore be given a value, be it arbitrary.
- For Radioss, planes will be output in Radioss ANIMATION files, but no contact computation is possible with Radioss FE in Radioss.
- The interfaces TYPE 14 and 15 allow to compute contact between MADYMO hyper-ellipsoids and Radioss FE. The option /MADYMO/LINK allows to connect a Radioss node to a MADYMO body. (See the Radioss CRASH input manual for a description of the options /MADYMO/LINK, /SURF/MDELLIPS, /INTER/TYPE14 and /INTER/TYPE15).

Input Example Basic Coupling

The following is part of a MADYMO input deck for Basic Coupling:

```
<COUPLING>
   <COUPLING_SURFACE
      EXTERNAL_REF="10"
      SURFACE="/2/1"
      EXTERNAL_DATA="0.0"/>
</COUPLING>
```

This will send data from surface /2/1 to Radioss. In Radioss one may refer to this data using reference 10. Note the arbitrary value 0.0 for EXTERNAL_DATA, which is not used in Radioss.
The following is part of a Radioss STARTER input deck for Basic Coupling:

```
/SURF/MDELLIPS/1001/The first madymo coupl'd surface
      10
```

Verification

The user is advised to check the log file and the reprint file to verify if all entities are communicated correctly. The log file shows the Radioss output of all entities received from MADYMO, the reprint file shows the entities MADYMO sends to Radioss.

The Radioss STARTER log file:

```
 EXTERNAL COUPLING TO SURFACE:    1001 The first madymo coupl'd surface
 ----------------------------

          SURFACE DEFINITION :
          --------------------
          REFERENCE TO MADYMO COUPLING SURFACE :       10
          CORRESPONDING DATA WILL BE READ IN RADIOSS ENGINE.
```

The accompanying MADYMO reprint file:

```
---------------------------------- COUPLING ----------------------------------
 COUPLING
    FE_MODEL .........................:
      SURFACE ........................: /2/1 ( /S2_first_ball/ell1_first_ball )
    EXTERNAL REFERENCE ...............:         10
    EXTERNAL DATA ....................:
    NUMBER ..:          1  DATA ..:      1.0000E+03
-------------------------------- END COUPLING --------------------------------
```

8.3.2 Extended Coupling

- MADYMO and Radioss extended coupling is available with Block Format 4.4 for Radioss STARTER with additional functionalities.
- The following additional option is available in Radioss version 4.7:

  ```
  /MADYMO/EXFEM/exfem_id/exfem_title
  ```

  in which exfem_id is the FEM identification number and exfem_title is the FEM name. Following this option, one has to specify the part id's of the elements of which information is exchanged with MADYMO:

  1 to n CARDS:

  ```
  1         2          3       ...
  Ip1       Ip2        Ip3     ...
  ```

  in which Ip1, Ip2, etc are the integer part id's.

  - exfem_id and exfem_title are not exchanged with MADYMO.
  - Part IDs, as well as related nodes IDs and elements IDs can be used in the MADYMO input file, in order to define contact interfaces between these Radioss entities and MADYMO MB or FE models. These parts must be parts of shells, 3 nodes shells or 8 nodes bricks.
  - Options /MADYMO/LINK cannot use any node belonging to these parts. Radioss STARTER will generate an error in such a case.

– A node belonging to such a part can not be a slave node of an interface TYPE 2, nor of a rigid body within Radioss, if it receives contact forces from MADYMO contact interfaces: Radioss STARTER writes a warning if a node belonging to the exchanged parts is a slave node of an interface TYPE 2 or of a rigid body.

Input Example Extended Coupling

The following is part of a MADYMO input deck.

```
<COUPLING
   FE_MODEL="/1/100"
>
```

The following is part of a Radioss input deck.

```
/MADYMO/EXFEM/1/
       2
```

This forces Radioss to send all information of nodes and elements belonging to part 2 to MADYMO.

Verification

The user is advised to check the log file and the reprint file to verify if all entities are communicated correctly. The Radioss log file shows the Radioss output of all FE data send to MADYMO, the reprint file shows the FE data that MADYMO received from Radioss.
Example Radioss log file:

```
      FEM INTERFACED TO MADYMO DEFINITION
      ----------------------------------


      FOLLOWING PARTS         WILL BE SENT TO MADYMO
       2

      FOLLOWING 4-NODES SHELL WILL BE SENT TO MADYMO
    4427    4428    4429    4430

      FOLLOWING 3-NODES SHELL WILL BE SENT TO MADYMO


      FOLLOWING BRICKS        WILL BE SENT TO MADYMO


      FOLLOWING NODES         WILL BE SENT TO MADYMO
    4308    4309    4310    4311    4312    4313    4314    4315    4316
```

Accompanying reprint file:

```
----------------------------COUPLING INTERFACE------------------------------
 NUMBER OF NODES                       :          9
 NUMBER OF PARTS                       :          1
   PART ID (FROM COUPLED PROGRAM)      :          2
     THICKNESS                         :      8.1000E-04
     BULK MODULUS                      :      6.9608E+10
 NUMBER OF ELEMENTS                    :          4
   NUMBER OF TRIADS                    :          0
   NUMBER OF QUADS                     :          4
   NUMBER OF SOLIDS                    :          0
```

8.4 Restart

For more general information on restart, see the Radioss engine manual and the MADYMO Reference Manual.

To enable restart, the following steps must be taken:

1. Generate restart files for Radioss and MADYMO.
2. Generate a Radioss engine input deck (D02 file). In the D02 file, set the identifier to `/RUN/ROOTNAME/<restart_id>`. Change the <restart_id> to select the desired Radioss restart file. Radioss restart files are cyclic. The number of Radioss restart files is determined by the `/RFILE/n` option in the engine file, where `n` determines the number of restart files.
3. Generate a MADYMO restart input file (xml file). There is a template restart file in `<madymo_dir>/etc/etc/template\_restart.xml`.
4. Submit the restart job with the `-rst <restart.rst>` command line option, where <restart.rst> points to the restart file generated in step 1. For an example restart startup script, see section 9.4.3

8.4.1 Restart Control

- All remarks for normal coupling to Radioss are valid for restart (see section 8.2).
- The MADYMO restart file is cumulative. All time points for Radioss are written in one file.
- The restart frequency if fully MADYMO controlled, and is enabled by the TIME_RESTART keyword in the MADYMO input deck. The TIME_RESTART and D01

RFILE don't need to be synchronized. MADYMO will write a restart file every TIME_RESTART seconds, and Radioss follows this.

- Please check the log file that both Radioss and MADYMO restart at the same time-point. Selecting the wrong Radioss restart file (see above) can result in out-of-sync restart runs.
- The restart analysis starts at the first timepoint saved in the restart file that is smaller than the begin time of the restart analysis.

8.5 Limitations and Known Issues

- The last ANIMATION file may not be written if the simulation end time is a multiple of the output time step. This situation can be avoided by increasing the simulation end time point slightly.
- Running a coupling executable between Radioss and MADYMO is not possible through the RADTOOL.

9 Example Startup Scripts

This section shows some examples to run coupling executables with LS-DYNA, MPP-DYNA, PAM CRASH / SAFE and Radioss. In general, the coupling is started with the following command line:

```
madymocli -coupling direct <input.xml> -x /path/to/coupling_exe -arg <partner args>
```

where <input.xml> is the MADYMO input deck, / path / to / coupling_exe is the PARTNER coupling executable and <partner args> are all arguments needed by this coupling executable.

The benefit of this approach is that MADYMO handles all environment variables that MADYMO needs to run in coupling, including adapting the run time library search path ($LIBPATH, $LD_LIBRARY_PATH or $SHLIB_PATH, depending on the OS).

9.1 PAM CRASH/SAFE

```
#!/bin/sh
# example script to run MADYMO/PAM coupling
# use with options: input.xml input.pc #cpu
#
PAMEXE=/path/to/pam2006/psolid.x
# PAMLIB is usually a subdir of the dir where psolid.x resides
PAMLIB=/path/to/pam2006/lib
XML=$1
PC=$2
CPU=$3
export OMP_NUM_THREADS=${CPU}
export PAMLIC=CRASHSAF
export LD_LIBRARY_PATH=${PAMLIB}
# assuming madymo64 is in the search path
madymo64 -coupling direct -nrproc ${CPU} -x ${PAMEXE} ${XML} -arg ${PC}
```

9.2 LS-DYNA

```
#!/bin/sh
# example script to run MADYMO/LSDYNA coupling
# use with options: <input.xml> <input.key>
XML=$1
K=$2
# lsdyna executable
LSEXE=/path/to/ls970.madymo64
# assuming madymo64 is in the search path
madymo64 -coupling direct $XML -x ${LSEXE} -arg y=madymo i="$K"
```

9.3 MPP-DYNA

9.3.1 MPP-DYNA with lammpi

```
#!/bin/sh
# example script to run MADYMO/MPPDYNA coupling
# use with options: <input.xml> <input.key> <#nodes for mppdyna>
XML=$1
K=$2
NP=$3
MPPEXE=/path/to/mpp971.madymo64
# make sure lammpi is in the path
PATH=/usr/local/mpi/lam-6.5.9/bin:${PATH};  export PATH
#
# build wrapper script for lam
# so that MADYMO env.vars are known at boot-time of the the lam universe
cat > ./lamrun.$$ << _EOF_
#!/bin/sh
lamboot -v
mpirun -np $NP $MPPEXE y=madymo i=$K $XML
wipe -v
_EOF_
#
chmod 755 ./lamrun.$$
# assuming madymo64 is in the search path
madymo64 -nopretrans -coupling direct $XML -x ./lamrun.$$
# cleanup
rm -f ./lamrun.$$
```

9.3.2 MPP-DYNA with mpich/native mpi

```
#!/bin/sh
# example script to run MADYMO/MPPDYNA coupling
# use with options: <input.xml> <input.key> <#nodes for mppdyna>
XML=$1
K=$2
NP=$3
# mppdyna executable
MPPEXE=/path/to/mpp971.madymo64
MPIRUN=/opt/mpi/bin/mpirun
# assuming madymo64 is in the search path
madymo64 -coupling direct $XML -x $MPIRUN -arg -np $NP $MPPEXE i=$K y=madymo
```

9.3.3 MPP-DYNA with mpi that doens’t propagate env. vars

```
#!/bin/sh
# example script to run MADYMO/MPPDYNA coupling
# use with options: <input.xml> <input.key> <#nodes for mppdyna>
XML=$1
K=$2
NP=$3
MPPEXE=/path/to/mpp971.madymo64
# make sure lammpi is in the path
```

```
MPIRUN=/usr/local/mpi/bin/mpirun
#
# build wrapper script for mpi
# to force setting env.vars before starting executable
cat > ./mpirun.$$ << _EOF_
#!/bin/sh
# wrapper script for dumb mpi
export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH
export MADETCPATH=\$MADETCPATH
export MADINCPATH=\$MADINCPATH
export MADBINPATH=\$MADBINPATH
mpirun -np $NP $MPPEXE y=madymo i=$K $XML
_EOF_
#
chmod 755 ./mpirun.$$
# assuming madymo64 is in the search path
madymo64 -nopretrans -coupling direct $XML -x ./mpirun.$$
# cleanup
rm -f ./mpirun.$$
```

9.3.4 MPP-DYNA with LSF on IBM

```
#!/bin/sh
# example script to run MADYMO/MPPDYNA coupling
# use with options: <input.xml> <input.key>
XML=$1
K=$2
# mppdyna executable
MPPEXE=/path/to/mpp971.madymo64
# start coupling through LSF pam
# assuming madymo64 is in the search path
madymo64 -coupling direct $XML -x pam -arg -g -1 poejob $MPPEXE i=$K y=madymo
```

9.4 Radioss

9.4.1 Radioss smp

```
#!/bin/sh
# example script to run MADYMO/Radioss coupling
# use with args: <input.xml> <input> (without extension!)
XML=$1
D00=$2.D00
D01=$2.D01
if ! [ -h file_engine ]
then
#   change this to make a softlink from the correct file_engine file
  ln -s  /path/to/staff/file_engine file_engine
fi
# set STARTER to Radioss starter
export STARTER=/path/to/radioss_exe/s474x_spmd
# set RADMAD to the coupled executable
export RADMAD=/path/to/rad2md/d474x_m64_pmd
#
```

```
${STARTER} < $D00
#
# assuming madymo64 is in the search path
madymo64  -coupling direct -x ${RADMAD} $1 < $D01
```

9.4.2 Radioss mpp

```
#!/bin/sh
# use with args: <input.xml> <input.D00> <input.D01> (including extension!)
XML=$1
D00=$2
D01=$3
if ! [ -h file_engine ]
then
#   change this to make a softlink from the correct file_engine file
  ln -s  /path/to/staff/file_engine file_engine
fi
# set STARTER to radioss starter
export STARTER=/path/to/radioss_exe/s474x_spmd
# set RADMAD to the coupled executable
export RADMAD=/path/to/rad2md/d474x_m64_pmd
#
${STARTER} < $D00
#
# assuming madymo64 is in the search path
madymo64 -coupling direct -x ${RADMAD} $XML -arg -p4pg hostsfile -p4wd ‘pwd‘ < $D01
```

9.4.3 Radioss mpp and restart

```
#!/bin/sh
# use with args: <input.xml> <restart.rst> <input.D02> (including extension!)
XML=$1
RST=$2
D02=$3
if ! [ -h file_engine ]
then
#   change this to make a softlink from the correct file_engine file
  ln -s  /path/to/staff/file_engine file_engine
fi
# starter not needed, allready done in normal run
# set RADMAD to the coupled executable
export RADMAD=/path/to/rad2md/d474x_m64_pmd

# assuming madymo64 is in the search path
# the next line must be on one line, but maybe textwrapping changes this
madymo64 -coupling direct -rst ${RST} -x ${RADMAD} $XML -arg -p4pg hostsfile -p4wd
‘pwd‘ < $D02
```

10 MATLAB/Simulink coupling

MADYMO provides a coupling with MATLAB/ Simulink. The coupling is available on platforms which are supported by both MADYMO and The MathWorks®, MATLAB R2007b (see the MathWorks User guide or visit www.mathworks.com). This does not exclude possible compatibility with another MATLAB version.

This coupling is based on a TCP/IP connection between the S-function named **madymo3d_server** and MADYMO. It allows the user to run MATLAB/ Simulink on the same or different computer than where MADYMO is running. However, note that MADYMO can only connect and communicate with the same type of platform using the IP address HOST_ADDRESS. See Fig. 10.1 for the detailed structure of the coupling of the two environments.

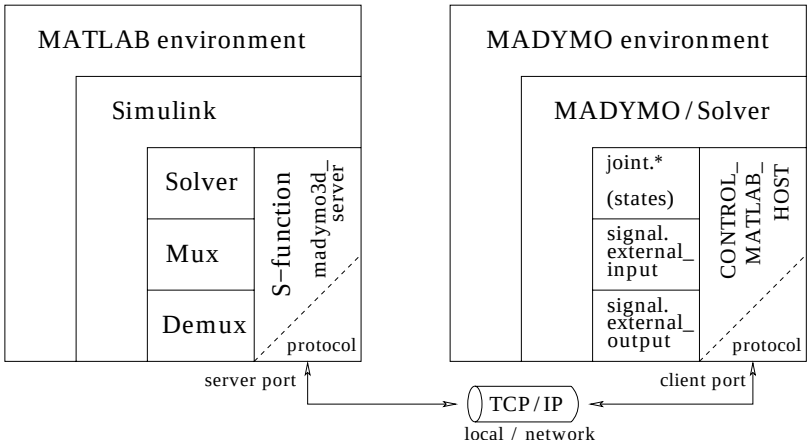


Figure 10.1: Detailed outline of the coupling.

10.1 Compatible MATLAB version

As explained, the S-function **madymo3d_server** is a coupling program running within MATLAB to communicate with MADYMO. For each program the system requirements are given by its vendor. For system requirements of the MADYMO solver see the Installation Instructions. Since we have two programmes, the system requirements may be conflicting. The coupling program of this MADYMO version is based on the MATLAB build and test versions are stated in the following table.

Solver Platformid	MATLAB build	MATLAB test
linux24-em64t	R14SP3	R16
em64t-win	R2007b	R2007b
linu24-x86	R14SP3	R2007b
win32p	R2007b	R2007b
linux24-x86_64	R14SP3	R16
linux26-x86_64	R14SP3	R2007b

The coupling is tested using MATLAB version R2007b on newer platforms. Known compatibility problems are noted on the linux operating systems. Depending on the linux version the coupling either can or can not run, if this is the case a different version of the linux operating system, MADYMO or MATLAB is needed.

10.2 Interfacing MADYMO with MATLAB/ Simulink

The interface between MATLAB/ Simulink and MADYMO is defined in both the Simulink and the MADYMO model. To set up a coupled simulation, perform the following steps:

1. Link both input and output ports by defining SIGNAL.EXTERNAL_INPUT and SIGNAL.EXTERNAL_OUTPUT in the MADYMO model to Mux and Demux blocks in the Simulink model file.
2. Select the time integration method 'MATLAB' / 'EULER' in the MADYMO model.
3. The simulation time and solver options in the Simulink model are selected via the Simulink graphical user interface shown in Fig. 10.2.
4. The integration time step should be set equal in both programmes.

All MADYMO functionality is available within MATLAB/ Simulink by means of a S-function called **madymo3d_server**. One or more instances of such block(s) can be defined in a Simulink model. An S-function is shown in Fig. 10.3, in which the parameters represent the following:

1. Command to be executed when the Simulink model is started. This character string normally ends with the MADYMO model XML-filename.
2. Number of 'continuous' states to be integrated by the MATLAB Solver, using INT_MTH="MATLAB" under CONTROL_ANALYSIS.TIME.
   This number should be greater than or equal to the maximum number of first

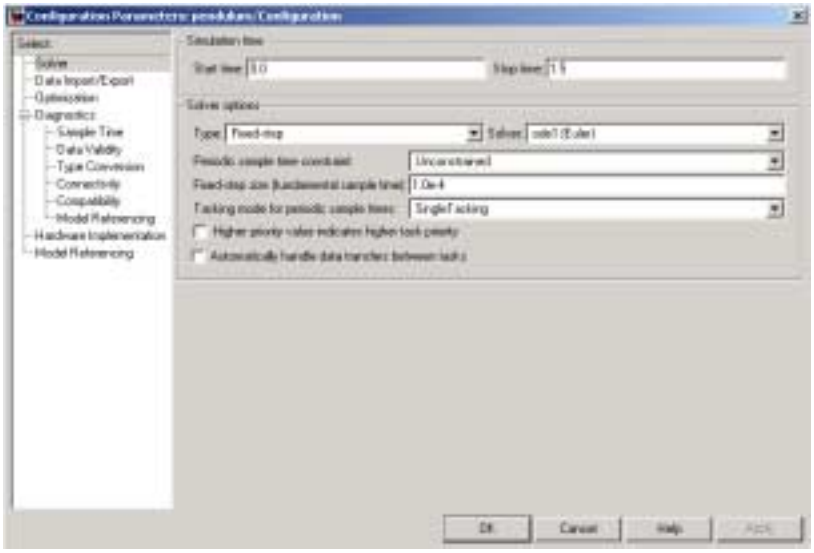


Figure 10.2: Example of the Configuration Parameters.

order differential equations in the MADYMO model.
Select '0' when the states are integrated by MADYMO (INT_MTH="EULER").

3. Input port dimension, largest EXTERNAL_REF defined for SIGNAL.EXTERNAL_-INPUT. The value -1 represents dynamically sized.
4. Output port dimension, largest EXTERNAL_REF defined for SIGNAL.EXTERNAL_OUTPUT. The value -1 represents dynamically sized.
5. Server port number, value of HOST_PORT. MADYMO connects to this port.
6. Polling time [s] for the server to wait on the connection with MADYMO.
7. Integration time step, value of TIME_STEP defined for CONTROL_ANALYSIS.-TIME.

10.3 Using MATLAB/ Simulink in combination with MADYMO

Open the MADYMO environment by typing

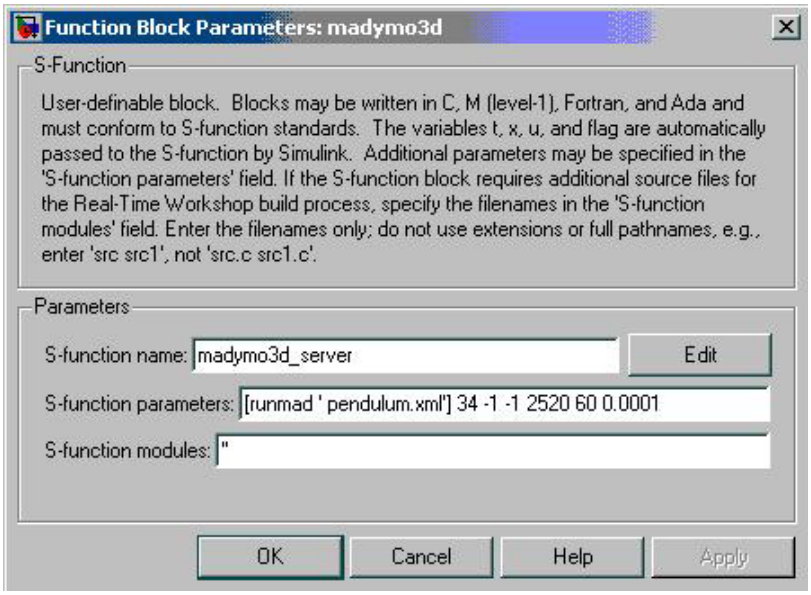

Function Block Parameters: madymo3d
S-Function
User-definable block. Blocks may be written in C, M (level-1), Fortran, and Ada and must conform to S-function standards. The variables t, x, u, and flag are automatically passed to the S-function by Simulink. Additional parameters may be specified in the 'S-function parameters' field. If the S-function block requires additional source files for the Real-Time Workshop build process, specify the filenames in the 'S-function modules' field. Enter the filenames only; do not use extensions or full pathnames, e.g., enter 'src src1', not 'src.c src1.c'.
Parameters
S-function name: madymo3d_server
Edit
S-function parameters: [runmad ' pendulum.xml'] 34 -1 -1 2520 60 0.0001
S-function modules: ''
OK
Cancel
Help
Apply

Figure 10.3: Example of the S-function settings.

madymo72 -sh

This is only needed when MADYMO specific environment variables are used in the m-files.

An example of the coupling, see Application Manual, is as follows:

- Open a command window (using either cmd or xterm).
- Go to the directory where the MADYMO model, the MATLAB/ Simulink model and the startup.m files are located.
- Start MATLAB from the command window.
- The file startup.m is executed automatically.
- The prompt is changed into '>>' indicating that MATLAB is running.
- Load or create the associated Simulink model, for example the pendulum shown in Fig. 10.4, containing an S-function named **madymo3d_server** and appropriate parameters shown in Fig. 10.3. Here, the MATLAB variable named 'runmad' refers to the command to execute MADYMO.

- Go to Simulation ⟶ Start in the Simulink graphical user interface to start the simulation.
- Parameters or models can be altered and the simulation can be started again.
- Note that one can also edit the startup.m file by replacing the environment variable `MADHOME` which the path to the madymodir. This gives you the ability to start MATLAB by clicking on the icon. It is good practice to store the startup file in the directory where the models are located.

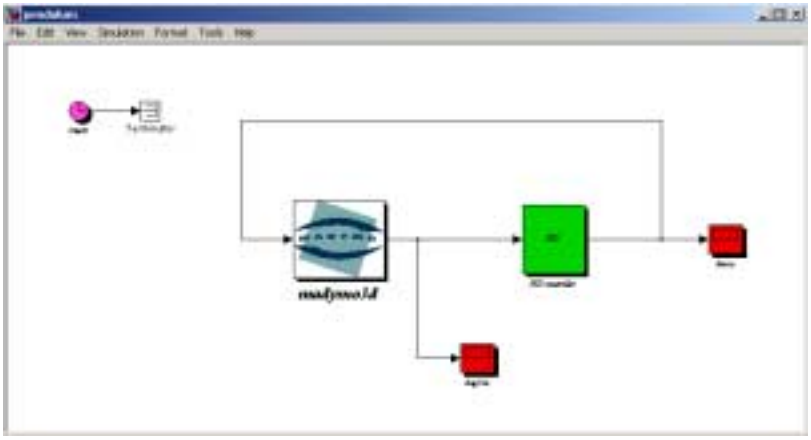


Figure 10.4: Example of the pendulum model

The coupled simulation runs until either the end time in the MADYMO model is reached, the end time in the Simulink model is reached or the simulation is terminated by the user.

10.4 Conditions

Accessing MADYMO from MATLAB/ Simulink via the S-function **madymo3d_server** is straight forward if the following is taken into account:

- MATLAB executes the m-file 'startup.m' when starting. The minimum list of commands is listed below.
- The coupling can run only if the MATLAB variable runmad is set and the search path includes the S-function **madymo3d_server**, version 7.2.

10.5 Miscellaneous

Using the coupling gives some side effects, mainly in the area of numerical stability. Most of the side effects concern the availability of the time integration and the computer platforms.

Here are some MATLAB / Simulink remarks:

- Development and testing takes place on the MATLAB Service Pack which was available at the MADYMO release date. It is most likely that newer Service Packs are working, where older Service Packs might not work on some platforms.
- Either ode1 (Euler) or ode2 (Heun) is the best MATLAB solver to start with for multi-body models. In some cases problems with ode4 are noted.
- The selection of the discrete (e.g. no continuous states) solver is the preferred choice to start with Finite Element models. Then, the states should be computed by the MADYMO implicit-explicit Euler integrator in MADYMO.
- The variable time step MATLAB (and MADYMO) solvers are not implemented for the MATLAB / Simulink coupling, because the integration time step is not updated in MADYMO.
- Only zero or one input / output ports are omitted. Both the input port width and the output port width should match the MADYMO model.
- During the termination phase of the Simulink model the platform 'win32p' and 'em64t-win' will continue, where the other platforms wait on the completion of the termination phase of MADYMO.
- For MATLAB running on Windows, batch processing can be used only if matlab.bat is started in combination with the option -wait. See the MATLAB manual pages for further documentation.
- The port used for the TCP / IP connection are not always available for the next simulation, since they might be locked by the system. The update time of the system administration can take quite some time.
- MADYMO information may be shown in the command screen of MATLAB during the coupled simulation. However all MADYMO related output files, including the LOG file are available separately.
- Errors in the MADYMO model will result in an error during the start of the Simulink model. An error message for the S-function block **madymo3d_server** will

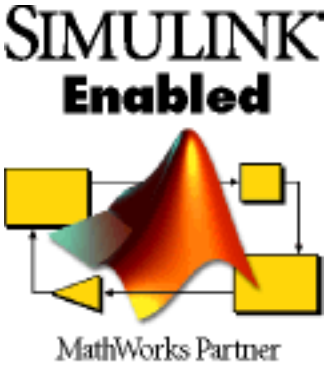
explain the type of error. For a detailed error message the LOG and / or REPRINT file should be examined.

- On linux, the MATLAB (environment) might cause abnormal termination during the parsing of the MADYMO input file. This is solved by setting the LD_PRELOAD environment variable of the gcc library.
  `setenv LD_PRELOAD /lib/libgcc_s.so.1`

Here are some MADYMO known issues:

- Since MATLAB leads the integration, the MB time step can not be changed during the simulation. The coupling with an external FE program is not supported.
- The stability of MADYMO is not only related to the parameters defined in the MADYMO model, but also related to the Simulink model when a closed control loop (feedback) is defined.
- Locking and unlocking by Coulomb friction, breaking of joints and mass belts are is not allowed when using the MATLAB solver, because no reset of the states is implemented. The MADYMO implicit-explicit Euler integrator can be used instead.
- When connecting, MADYMO expects the **madymo3d_server** to be running on the IP address HOST_ADDRESS. It might connect to another server application when the port on the host is providing another server application. In this case the MADYMO connects successfully, but subsequently waits or terminates.
- Coupling is available for computers running in SMP mode, but not in MPP mode.

See MADYMO Reference Manual, section CONTROL_MATLAB_HOST, the MATLAB and the Simulink User guide for further details. TASS BV is a partner of The MathWorks.


SIMULINK
Enabled
MathWorks Partner